Tabs

Tabs A comprehensive tab navigation component for organizing content into multiple panels. Features include multiple size variants (xs to xl), icon positioning, bordered styles, container backgrounds, custom styling, scrollable overflow, closeable tabs, and full accessibility support. How to use ■ import { AavaTabsComponent } from "@aava/play-core" ; Basic Usage ■ The Tabs component allows you to organize content into multiple panels, with only one visible at a time. This is ideal for dashboards, settings, or any interface where users need to switch between related views without navigating away from the page. Default behavior: Horizontal layout, first tab active, content panel below. Usage: Pass an array of tab objects to the tabs input. Each tab can have a label, content, and optional properties. Angular Preview Code < aava-tabs [tabs] = " [{ id: 'tab1', label: 'Tab 1', content: 'Content 1' }, { id: 'tab2', label: 'Tab 2', content: 'Content 2' }, { id: 'tab3', label: 'Tab 3', content: 'Content 3' }] " activeTabId = " tab1 " > </ aava-tabs > Variants ■ Tabs support multiple visual variants for different use cases: Default: Standard tabbed navigation with underline indicator and optional icons Button: Tabs appear as pill or rounded buttons with optional borders and custom styling Icon: Tabs display icons with labels, supporting start/end icon positioning Icon Only Square: Compact square tabs showing only icons, ideal for toolbars Icon Only Circle: Circular icon-only tabs for minimal interfaces Choose the variant that best fits your design and use case. The variant input controls the style, with additional options like bordered , iconPosition , and buttonShape for further customization. Angular Preview Code <!-- Default variant with underline --> < aava-tabs [tabs] = " [{ id: 'default1', label: 'Home', iconName: 'home', content: 'Welcome to the home page' }, { id: 'default2', label: 'About', iconName: 'info', content: 'Learn more about us' }, { id: 'default3', label: 'Contact', iconName: 'phone', content: 'Get in touch' }] " variant = " default " activeTabId = " default1 " [showContentPanels] = " false " > </ aava-tabs > <!-- Button variant - Pill with Border --> < aava-tabs [tabs] = " [{ id: 'pill1', label: 'Overview', iconName: 'layout', content: 'Overview content' }, { id: 'pill2', label: 'Analytics', iconName: 'chart-no-axes-combined', content: 'Analytics content' }, { id: 'pill3', label: 'Reports', iconName: 'file-text', content: 'Reports content' }] " variant = " button " buttonShape = " pill " [bordered] = " true " size = " sm " activeTabId = " pill1 " [showContentPanels] = " false " > </ aava-tabs > <!-- Button variant - Rectangle with Border --> < aava-tabs [tabs] = " [{ id: 'rect1', label: 'Projects', iconName: 'folder', content: 'Projects content' }, { id: 'rect2', label: 'Tasks', iconName: 'check-square', content: 'Tasks content' }, { id: 'rect3', label: 'Settings', iconName: 'settings', content: 'Settings content' }] " variant = " button " buttonShape = " rounded " [bordered] = " true " size = " sm " activeTabId = " rect1 " [showContentPanels] = " false " > </ aava-tabs > <!-- Icon Only Square variant --> < aava-tabs [tabs] = " [{ id: 'icon1', iconName: 'home', label: 'Home', content: 'Home content' }, { id: 'icon2', iconName: 'users', label: 'Users', content: 'Users content' }, { id: 'icon3', iconName: 'settings', label: 'Settings', content: 'Settings content' }] " variant = " iconOnlySquare " activeTabId = " icon1 " [showContentPanels] = " false " > </ aava-tabs > Size

Variants ■ Tabs support five size variants to fit different interface densities and use cases: Extra Small (xs): Compact tabs for dense interfaces or mobile layouts Small (sm): Smaller tabs for secondary navigation or sidebar use Medium (md): Default size, balanced for most use cases Large (lg): Larger tabs for primary navigation or desktop interfaces Extra Large (xl): Maximum size for prominent navigation or accessibility needs Choose the size that best matches your interface hierarchy and available space. The size input controls the overall scale of tabs, icons, and spacing. Angular Preview Code <!-- Extra Large Size --> < aava-tabs [tabs] = " [{ id: 'xl1', label: 'Overview', iconName: 'layout', content: 'XL Overview Content' }, { id: 'xl2', label: 'Analytics', iconName: 'chart-no-axes-combined', content: 'XL Analytics Content' }, { id: 'xl3', label: 'Reports', iconName: 'file-text', content: 'XL Reports Content' }] " variant = " button " buttonShape = " rounded " size = " xl " [bordered] = " true " activeTabId = " xl1 " [showContentPanels] = " false " > </ aava-tabs > <!-- Large Size --> < aava-tabs [tabs] = " [{ id: 'lg1', label: 'Overview', iconName: 'layout', content: 'LG Overview Content' }, { id: 'lg2', label: 'Analytics', iconName: 'chart-no-axes-combined', content: 'LG Analytics Content' }, { id: 'lg3', label: 'Reports', iconName: 'file-text', content: 'LG Reports Content' }] " variant = " button " buttonShape = " rounded " size = " lg " [bordered] = " true " activeTabId = " lg1 " [showContentPanels] = " false " > </ aava-tabs > <!-- Medium Size (Default) --> < aava-tabs [tabs] = " [{ id: 'md1', label: 'Overview', iconName: 'layout', content: 'MD Overview Content' }, { id: 'md2', label: 'Analytics', iconName: 'chart-no-axes-combined', content: 'MD Analytics Content' }, { id: 'md3', label: 'Reports', iconName: 'file-text', content: 'MD Reports Content' }] " variant = " button " buttonShape = " rounded " size = " md " [bordered] = " true " activeTabId = " md1 " [showContentPanels] = " false " > </ aava-tabs > <!-- Small Size --> < aava-tabs [tabs] = " [{ id: 'sm1', label: 'Overview', iconName: 'layout', content: 'SM Overview Content' }, { id: 'sm2', label: 'Analytics', iconName: 'chart-no-axes-combined', content: 'SM Analytics Content' }, { id: 'sm3', label: 'Reports', iconName: 'file-text', content: 'SM Reports Content' }] " variant = " button " buttonShape = " rounded " size = " sm " [bordered] = " true " activeTabId = " sm1 " [showContentPanels] = " false " > </ aava-tabs > <!-- Extra Small Size --> < aava-tabs [tabs] = " [{ id: 'xs1', label: 'Overview', iconName: 'layout', content: 'XS Overview Content' }, { id: 'xs2', label: 'Analytics', iconName: 'chart-no-axes-combined', content: 'XS Analytics Content' }, { id: 'xs3', label: 'Reports', iconName: 'file-text', content: 'XS Reports Content' }] " variant = " button " buttonShape = " rounded " size = " xs " [bordered] = " true " activeTabId = " xs1 " [showContentPanels] = " false " > </ aava-tabs > Tabs with Icons ■ Add icons to tabs using the iconName property in tab objects. Control icon positioning with the iconPosition input for flexible layouts. Icon Position: Use iconPosition="start" (default) or iconPosition="end" to control icon placement Icon-Only Variants: Use iconOnlySquare or iconOnlyCircle variants for compact, icon-only tabs When to use: For quick recognition, navigation clarity, or when space is limited Angular Preview Code <!-- Icons at start (default) --> < aava-tabs [tabs] = " [{ id: 'home1', label: 'Home', iconName: 'home', content: 'Welcome home!' }, { id: 'user1', label: 'Profile', iconName: 'user', content: 'Your profile' }, { id: 'settings1', label: 'Settings', iconName: 'settings', content: 'App settings' }] " iconPosition = " start " activeTabId = " home1 " [showContentPanels] = " false " > </ aava-tabs > <!-- Icons at end --> < aava-tabs [tabs] = " [{ id: 'home2', label: 'Home', iconName:

'chevron-right', content: 'Welcome home!' }, { id: 'user2', label: 'Profile', iconName: 'chevron-right', content: 'Your profile' }, { id: 'settings2', label: 'Settings', iconName: 'chevron-right', content: 'App settings' }] " iconPosition = " end " activeTabId = " home2 " [showContentPanels] = " false " > </ aava-tabs > <!-- Icon Only Square --> < aava-tabs [tabs] = " [{ id: 'home3', label: 'Home', iconName: 'home', content: 'Welcome home!' }, { id: 'user3', label: 'Profile', iconName: 'user', content: 'Your profile' }, { id: 'settings3', label: 'Settings', iconName: 'settings', content: 'App settings' }] " variant = " iconOnlySquare " activeTabId = " home3 " [showContentPanels] = " false " > </ aava-tabs > Tabs with Badges ■ Tabs can display badges to indicate counts, notifications, or status. Add a badge property to any tab object. How: Set badge to a number or string. When to use: For unread counts, alerts, or status indicators. Angular Preview Code < aava-tabs [tabs] = " [{ id: 'tab1', label: 'Inbox', badge: 5, content: 'Inbox Content' }, { id: 'tab2', label: 'Notifications', badge: 2, content: 'Notifications Content' }, { id: 'tab3', label: 'Archive', content: 'Archive Content' }] " activeTabId = " tab1 " > </ aava-tabs > Scrollable Tabs ■ For interfaces with many tabs, enable scrollable to handle overflow. Scrollable tabs show left/right arrows. Scrollable: Use for wide tab sets where horizontal space is available. Angular Preview Code < aava-tabs [tabs] = " [{ id: 'tab1', label: 'Tab 1', content: 'Content 1' }, { id: 'tab2', label: 'Tab 2', content: 'Content 2' }, { id: 'tab3', label: 'Tab 3', content: 'Content 3' }, { id: 'tab4', label: 'Tab 4', content: 'Content 4' }, { id: 'tab5', label: 'Tab 5', content: 'Content 5' }, { id: 'tab6', label: 'Tab 6', content: 'Content 6' }] " scrollable = " true " maxVisibleTabs = " 4 " activeTabId = " tab1 " > </ aava-tabs > Vertical Tabs ■ Switch to vertical orientation for side navigation or settings panels. Set orientation="vertical" . When to use: For sidebar navigation, settings, or when vertical space is preferred. Angular Preview Code Custom Styles ■ Tabs provide multiple styling options for comprehensive customization: Tab Row Styling: Use tabRowWrapperStyles and tabRowBackgroundStyles for custom tab container appearance Container Backgrounds: Enable showTabsContainerBg for fixed padding and background, or showcontainerBg for full container styling Custom CSS Properties: Use customStyles to override design tokens like --tabs-gap and color variables Active Button Styles: Customize active button appearance with activeButtonTabStyles for button variants Border Options: Add borders to button and icon variants using the bordered input These options allow complete visual customization while maintaining accessibility and functionality. Angular Preview Code API Reference ■ Inputs ■ Property Type Default Description tabs TabItem[] [] Array of tab objects (see TabItem interface) activeTabId string '' ID of the currently active tab variant 'default' | 'button' | 'icon' | 'iconOnlySquare' | 'iconOnlyCircle' 'default' Visual style of the tabs size 'xs' | 'sm' | 'md' | 'lg' | 'xl' 'md' Size of the tabs bordered boolean false Add border around tabs (for icon-only variants) iconPosition 'start' | 'end' 'start' Position of icons relative to text showcontainerBg boolean false Show background and border for entire tabs container showTabsContainerBg boolean false Show tabs container background with fixed padding customStyles Record<string, string> {} Custom styles for the tabs component disabled boolean false Disable all tabs scrollable boolean false Enable horizontal scrolling for overflow showDropdown boolean false Show dropdown for overflowed tabs maxVisibleTabs number 5 Max number of visible tabs before overflow allowTabClose boolean false Allow tabs to be closed centeredTabs boolean false Center the tabs

in the container fullWidth boolean false Tabs take full width of container animateTransitions boolean true Animate tab transitions lazyLoadContent boolean false Lazy load tab content persistActiveTab boolean true Persist active tab on tab list changes ariaLabel string 'Tabs navigation' ARIA label for accessibility activeButtonTabStyles Record<string, string> {} Custom styles for active button tab (button variant) buttonShape 'rounded' | 'pill' 'rounded' Shape for button variant showContentPanels boolean true Show content panels below tabs tabRowWrapperStyles Record<string, string> {} Custom styles for tab row wrapper tabRowBackgroundStyles Record<string, string> {} Custom styles for tab row background orientation 'horizontal' | 'vertical' 'horizontal' Orientation of the tabs TabItem Interface ■ interface TabItem { id : string ; label : string ; content ? : string ; iconName ? : string ; subtitle ? : string ; disabled ? : boolean ; badge ? : string | number ; closeable ? : boolean ; } Outputs ■ Event Type Description tabChange EventEmitter<TabItem> Emitted when the active tab changes tabClose EventEmitter<TabItem> Emitted when a tab is closed tabsReorder EventEmitter<TabItem[]> Emitted when tabs are reordered dropdownToggle EventEmitter<boolean> Emitted when the dropdown is toggled Methods ■ onTabClick(tab: TabItem): void — Activate a tab programmatically onTabClose(tab: TabItem, event: Event): void — Close a tab programmatically toggleDropdown(): void — Toggle the dropdown menu scrollLeft(): void / scrollRight(): void — Scroll the tab list Accessibility ■ The Tabs component follows WAI-ARIA accessibility guidelines and provides comprehensive keyboard navigation: Keyboard Navigation : Full support for Tab, Arrow keys (Left/Right/Up/Down), Enter, and Space ARIA Roles : Proper tablist , tab , and tabpanel roles for screen reader compatibility ARIA Attributes : aria-selected , aria-controls , aria-labelledby , and aria-disabled for state communication Focus Management : Visible focus indicators and proper tab sequence navigation Screen Reader Support : Descriptive labels, state announcements, and semantic HTML structure Keyboard Shortcuts : Arrow keys for tab navigation, Enter/Space for activation High Contrast : Enhanced borders and indicators for visibility in high contrast mode Design Tokens & Theming ■ AAVA Play Tabs use semantic design tokens for all surfaces, spacing, typography, and motion. The component exposes scoped override tokens for fine-tuning appearance while maintaining design system consistency. Available Design Tokens for Tabs ■ Size & Layout Tokens ■ Token Purpose Default Value --tabs-gap Spacing between tabs Theme-based Color & Visual Tokens ■ Token Purpose Default Value --tabs-tab-color Color for inactive tab text Theme-based --tabs-tab-active-color Color for active tab text Theme-based --tabs-tab-disabled-color Color for disabled tab text Theme-based Badge & Icon Tokens ■ Token Purpose Default Value --tabs-icon-size Default size for tab icons Theme-based Best Practices ■ Design Guidelines ■ Semantic Organization : Group related content logically and use clear, descriptive tab labels Tab Count : Limit to 5-7 tabs for optimal usability; use scrollable or dropdown for more Variant Selection : Choose variants based on context (default for content, button for actions, icon-only for compact) Size Selection : Use appropriate sizes (xs-xl) based on interface hierarchy and available space Icon Positioning : Consider iconPosition="end" for navigation arrows or action indicators Border Usage : Apply bordered input for better visual separation in dense interfaces Container Backgrounds : Use showTabsContainerBg or

showcontainerBg for visual grouping Visual Hierarchy : Use active states, badges, and icons to guide user attention appropriately Responsive Design : Consider vertical orientation or dropdown behavior for mobile devices Consistent Sizing : Match tab sizes to surrounding interface elements and content density Accessibility ■ Meaningful Labels : Provide clear, descriptive labels that explain each tab's content or purpose Unique IDs : Ensure each tab has a unique ID for proper ARIA relationships Keyboard Support : Test complete keyboard navigation flow including arrow keys and activation Screen Reader Testing : Verify proper announcement of tab states, content, and navigation Focus Indicators : Maintain clear visual focus states for keyboard navigation Color Contrast : Ensure sufficient contrast for all tab states and variants Reduced Motion : Respect user preferences for reduced motion in transitions Performance ■ Lazy Loading : Enable lazyLoadContent for tabs with heavy content to improve initial load Content Optimization : Only render active tab content unless pre-loading is necessary Icon Efficiency : Use icon systems efficiently and avoid loading unnecessary icon variants Smooth Scrolling : Use hardware-accelerated scrolling for large tab sets Event Handling : Debounce rapid tab switching and optimize event listeners Bundle Optimization : Import only needed variants and features to minimize bundle size Memory Management : Properly clean up event listeners and subscriptions in closeable tabs

```
<aava-tabs
  [tabs]="[
    { id: 'tab1', label: 'Tab 1', content: 'Content 1' },
    { id: 'tab2', label: 'Tab 2', content: 'Content 2' },
    { id: 'tab3', label: 'Tab 3', content: 'Content 3' }
  ]"
  activeTabId="tab1"
></aava-tabs>
```

```html
<!-- Default variant with underline -->
<aava-tabs
  [tabs]="[
    { id: 'default1', label: 'Home', iconName: 'home', content: 'Welcome to the home page' },
    { id: 'default2', label: 'About', iconName: 'info', content: 'Learn more about us' },
    { id: 'default3', label: 'Contact', iconName: 'phone', content: 'Get in touch' }
  ]"
  variant="default"
  activeTabId="default1"
  [showContentPanels]="false"
></aava-tabs>

<!-- Button variant - Pill with Border -->
<aava-tabs
  [tabs]="[
    { id: 'pill1', label: 'Overview', iconName: 'layout', content: 'Overview content' },
    { id: 'pill2', label: 'Analytics', iconName: 'chart-no-axes-combined', content: 'Analytics content'
    { id: 'pill3', label: 'Reports', iconName: 'file-text', content: 'Reports content' }
  ]"
  variant="button"
  buttonShape="pill"
  [bordered]="true"
  size="sm"
  activeTabId="pill1"
  [showContentPanels]="false"
></aava-tabs>

<!-- Button variant - Rectangle with Border -->
<aava-tabs
  [tabs]="[
    { id: 'rect1', label: 'Projects', iconName: 'folder', content: 'Projects content' },
    { id: 'rect2', label: 'Tasks', iconName: 'check-square', content: 'Tasks content' },
    { id: 'rect3', label: 'Settings', iconName: 'settings', content: 'Settings content' }
  ]"
  variant="button"
  buttonShape="rounded"
  [bordered]="true"
  size="sm"
  activeTabId="rect1"
  [showContentPanels]="false"
></aava-tabs>

<!-- Icon Only Square variant -->
<aava-tabs
  [tabs]="[
    { id: 'icon1', iconName: 'home', label: 'Home', content: 'Home content' },
    { id: 'icon2', iconName: 'users', label: 'Users', content: 'Users content' },
    { id: 'icon3', iconName: 'settings', label: 'Settings', content: 'Settings content' }
  ]"
  variant="iconOnlySquare"
  activeTabId="icon1"
  [showContentPanels]="false"
></aava-tabs>
```

```
<!-- Extra Large Size -->
<aava-tabs
  [tabs]="[
    { id: 'xl1', label: 'Overview', iconName: 'layout', content: 'XL Overview Content' },
    { id: 'xl2', label: 'Analytics', iconName: 'chart-no-axes-combined', content: 'XL Analytics Content'
    { id: 'xl3', label: 'Reports', iconName: 'file-text', content: 'XL Reports Content' }
  ]"
  variant="button"
  buttonShape="rounded"
  size="xl"
  [bordered]="true"
  activeTabId="xl1"
  [showContentPanels]="false"
></aava-tabs>

<!-- Large Size -->
<aava-tabs
  [tabs]="[
    { id: 'lg1', label: 'Overview', iconName: 'layout', content: 'LG Overview Content' },
    { id: 'lg2', label: 'Analytics', iconName: 'chart-no-axes-combined', content: 'LG Analytics Content'
    { id: 'lg3', label: 'Reports', iconName: 'file-text', content: 'LG Reports Content' }
  ]"
  variant="button"
  buttonShape="rounded"
  size="lg"
  [bordered]="true"
  activeTabId="lg1"
  [showContentPanels]="false"
></aava-tabs>

<!-- Medium Size (Default) -->
<aava-tabs
  [tabs]="[
    { id: 'md1', label: 'Overview', iconName: 'layout', content: 'MD Overview Content' },
    { id: 'md2', label: 'Analytics', iconName: 'chart-no-axes-combined', content: 'MD Analytics Content'
    { id: 'md3', label: 'Reports', iconName: 'file-text', content: 'MD Reports Content' }
  ]"
  variant="button"
  buttonShape="rounded"
  size="md"
  [bordered]="true"
  activeTabId="md1"
  [showContentPanels]="false"
></aava-tabs>

<!-- Small Size -->
<aava-tabs
  [tabs]="[
    { id: 'sm1', label: 'Overview', iconName: 'layout', content: 'SM Overview Content' },
    { id: 'sm2', label: 'Analytics', iconName: 'chart-no-axes-combined', content: 'SM Analytics Content'
    { id: 'sm3', label: 'Reports', iconName: 'file-text', content: 'SM Reports Content' }
  ]"
  variant="button"
  buttonShape="rounded"
  size="sm"
  [bordered]="true"
```

```
  activeTabId="sm1"
  [showContentPanels]="false"
></aava-tabs>

<!-- Extra Small Size -->
<aava-tabs
  [tabs]="[
    { id: 'xs1', label: 'Overview', iconName: 'layout', content: 'XS Overview Content' },
    { id: 'xs2', label: 'Analytics', iconName: 'chart-no-axes-combined', content: 'XS Analytics Content'
    { id: 'xs3', label: 'Reports', iconName: 'file-text', content: 'XS Reports Content' }
  ]"
  variant="button"
  buttonShape="rounded"
  size="xs"
  [bordered]="true"
  activeTabId="xs1"
  [showContentPanels]="false"
></aava-tabs>


<!-- Icons at start (default) -->
<aava-tabs
  [tabs]="[
    { id: 'home1', label: 'Home', iconName: 'home', content: 'Welcome home!' },
    { id: 'user1', label: 'Profile', iconName: 'user', content: 'Your profile' },
    { id: 'settings1', label: 'Settings', iconName: 'settings', content: 'App settings' }
  ]"
  iconPosition="start"
  activeTabId="home1"
  [showContentPanels]="false"
></aava-tabs>

<!-- Icons at end -->
<aava-tabs
  [tabs]="[
    { id: 'home2', label: 'Home', iconName: 'chevron-right', content: 'Welcome home!' },
    { id: 'user2', label: 'Profile', iconName: 'chevron-right', content: 'Your profile' },
    { id: 'settings2', label: 'Settings', iconName: 'chevron-right', content: 'App settings' }
  ]"
  iconPosition="end"
  activeTabId="home2"
  [showContentPanels]="false"
></aava-tabs>

<!-- Icon Only Square -->
<aava-tabs
  [tabs]="[
    { id: 'home3', label: 'Home', iconName: 'home', content: 'Welcome home!' },
    { id: 'user3', label: 'Profile', iconName: 'user', content: 'Your profile' },
    { id: 'settings3', label: 'Settings', iconName: 'settings', content: 'App settings' }
  ]"
  variant="iconOnlySquare"
  activeTabId="home3"
  [showContentPanels]="false"
></aava-tabs>
```

```
<aava-tabs
  [tabs]="[
    { id: 'tab1', label: 'Inbox', badge: 5, content: 'Inbox Content' },
    { id: 'tab2', label: 'Notifications', badge: 2, content: 'Notifications Content' },
    { id: 'tab3', label: 'Archive', content: 'Archive Content' }
  ]"
  activeTabId="tab1"
></aava-tabs>


<aava-tabs
  [tabs]="[
    { id: 'tab1', label: 'Tab 1', content: 'Content 1' },
    { id: 'tab2', label: 'Tab 2', content: 'Content 2' },
    { id: 'tab3', label: 'Tab 3', content: 'Content 3' },
    { id: 'tab4', label: 'Tab 4', content: 'Content 4' },
    { id: 'tab5', label: 'Tab 5', content: 'Content 5' },
    { id: 'tab6', label: 'Tab 6', content: 'Content 6' }
  ]"
  scrollable="true"
  maxVisibleTabs="4"
  activeTabId="tab1"
></aava-tabs>
```

■ No code found

■ No code found